

Trabalho 01 - Linguagens de Programação

Gustavo Azevedo Naldoni e Lucas Fernandes Gafaria Alves

29 de setembro de 2023

1 Exercício 01

```
imprima([X])      :- write(X).  
imprima([A|B])    :- imprima([A]),  
                    imprima(B).
```

2 Exercício 02

```
verificanota([X]) :- X >= 0,  
                    X < 30,  
                    write(X), write(' Reprovado').  
  
verificanota([X]) :- X >= 30,  
                    X < 60,  
                    write(X), write(' Recuperação').  
  
verificanota([X]) :- X >= 60,  
                    X < 100,  
                    write(X), write(' Aprovado').  
  
verificanota([A|B]) :- verificanota([A]),  
                      verificanota(B).
```

3 Exercício 03

```
vogal(a).  
vogal(e).  
vogal(i).  
vogal(o).  
vogal(u).  
  
contavogal([], 0).
```

```

contavogal([A|B], Contador) :- vogal(A),
    contavogal(B, Contador1),
    Contador is Contador1 + 1.

contavogal([A|B], Contador) :- not(vogal(A)),
    contavogal(B, Contador).

```

4 Exercício 04

```

insere(Elem, Lista, [Elem|Lista]).

extenso(1, um).
extenso(2, dois).
extenso(3, tres).
extenso(4, quatro).
extenso(5, cinco).
extenso(6, seis).
extenso(7, sete).
extenso(8, oito).
extenso(9, nove).
extenso(10, dez).

traduza([], []).
traduza([A|B], L) :- extenso(A, R),
    insere(R, L1, L),
    traduza(B, L1).

```

5 Exercício 05

```

duplica([], []).
duplica([A|B], [A, A|C]) :- duplica(B, C).

```